FINAL REPORT
The Battle of the Neighbourhoods- PART 2
(Data Capstone Project)

Introduction-

Mumbai is a big fast moving city in India.It gives homes to millions and also acts as a tourist spot.Mumbai is known as the economical capital of india.There are many places to go around in the city. There are many restaurants and cafes in Mumbai that are famous for their different food items.There are many places for tourists to go shopping at.

Problems-

1)List and visualize all the restaurants and cafes in Mumbai.
2)List and visualize all the Hotels in Mumbai.
3)List and visualize all the shopping areas in Mumbai

Goal of the project-

The goal of this project is to create a representation for the tourists visiting mumbai in order to make their stay comfortable in this beautiful city.This representation aims to show them the restaurants,hotels and shopping areas to go visit and stay at.

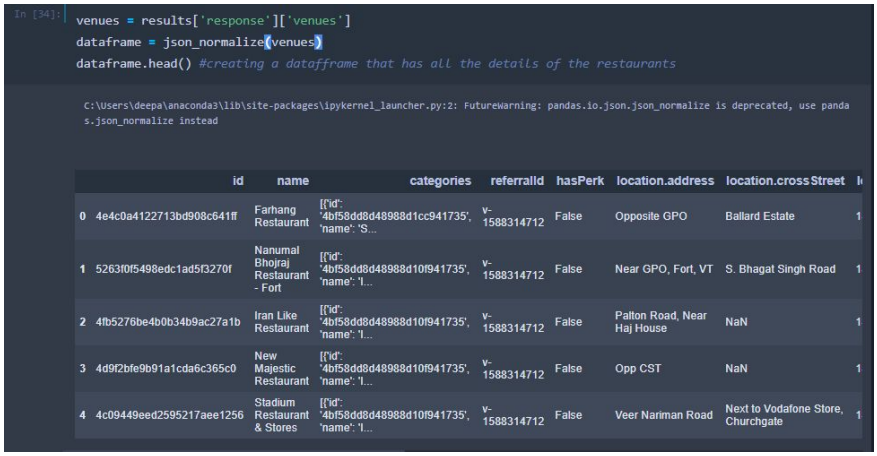
DATA SECTION-

To get all of this data, i am using the **Foursquare API** which is sufficient to provide me with all the data i need.It can provide me with the categories,names of the places and most importantly, its latitude and longitudes

METHODOLOGY-

Step1) We begin by importing all the libraries that we might need while doing this problem.Once we are done with that,we can put in our client id and the key to access the Foursquare API.

Step2) We put in the location in my case,Mumbai, and first we look for all the restaurants using the API. We fetch all the data and store it in a form of dataframe.

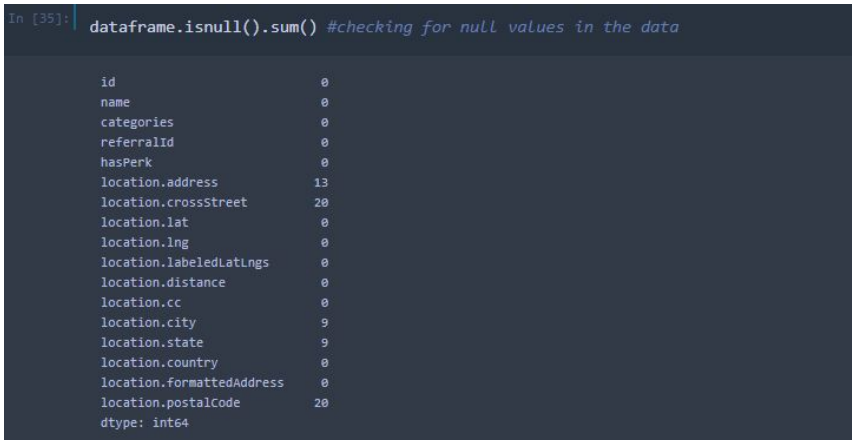


```
In [34]: venues = results['response']['venues']
         dataframe = json_normalize(venues)
         dataframe.head() #creating a datafframe that has all the details of the restaurants
```

```
C:\Users\deepa\anaconda3\lib\site-packages\ipykernel_launcher.py:2: FutureWarning: pandas.io.json.json_normalize is deprecated, use panda
s.json_normalize instead
```

	id	name	categories	referralId	hasPerk	location.address	location.crossStreet	l
0	4e4c0a4122713bd908c641ff	Farhang Restaurant	[{'id': '4bf58dd8d48988d1cc941735', 'name': 'S...	v-1588314712	False	Opposite GPO	Ballard Estate	1
1	5263f0f5498edc1ad5f3270f	Nanumal Bhojraj Restaurant - Fort	[{'id': '4bf58dd8d48988d10f941735', 'name': 'I...	v-1588314712	False	Near GPO, Fort, VT	S. Bhagat Singh Road	1
2	4fb5276be4b0b34b9ac27a1b	Iran Like Restaurant	[{'id': '4bf58dd8d48988d10f941735', 'name': 'I...	v-1588314712	False	Palton Road, Near Haj House	NaN	1
3	4d9f2bfe9b91a1cda6c365c0	New Majestic Restaurant	[{'id': '4bf58dd8d48988d10f941735', 'name': 'I...	v-1588314712	False	Opp CST	NaN	1
4	4c09449eed2595217aee1256	Stadium Restaurant & Stores	[{'id': '4bf58dd8d48988d10f941735', 'name': 'I...	v-1588314712	False	Veer Nariman Road	Next to Vodafone Store, Churchgate	1

Step3)We check for all the null values that might be present in our data before cleaning our data.



```
In [35]: dataframe.isnull().sum() #checking for null values in the data
```

```
id                        0
name                      0
categories                0
referralId                0
hasPerk                   0
location.address         13
location.crossStreet     20
location.lat              0
location.lng              0
location.labeledLatLngs   0
location.distance         0
location.cc               0
location.city             9
location.state            9
location.country          0
location.formattedAddress 0
location.postalCode      20
dtype: int64
```

Step4)We drop all the null values and drop all those columns which do not provide any value to our data.And the we then get the data that we can visualize.

```
In [41]:
clean_Restaurant_dataframe2= clean_Restaurant_dataframe.drop(['cc', 'city', 'country', 'crossStreet', 'form
                                    'labeledLatLngs', 'postalCode', 'id','state','distance'], axis=1)
clean_Restaurant_dataframe2 #dropping all the unwanted columnns from the dataframe
```

	name	categories	address	lat	lng
0	Farhang Restaurant	Steakhouse	Opposite GPO	18.938165	72.837917
1	Nanumal Bhojraj Restaurant - Fort	Indian Restaurant	Near GPO, Fort, VT	18.937974	72.837663
2	Iran Like Restaurant	Indian Restaurant	Patton Road, Near Haj House	18.945343	72.836809
3	New Majestic Restaurant	Indian Restaurant	Opp CST	18.938972	72.835517
4	Stadium Restaurant & Stores	Indian Restaurant	Veer Nariman Road	18.933173	72.826929
5	Sai Pooja Restaurant	None	NaN	18.938357	72.835073
6	Panchratna restaurant & bar	Indian Restaurant	Opp. Metro Cinema	18.942900	72.826820
7	Poonam Restaurant	Restaurant	NaN	18.939225	72.835707
8	Nanumal Bhojraj Restaurant - Masjid Bunder	Indian Restaurant	Gaumukh Bhawan, Masjid Bunder West, Near Masji...	18.953200	72.837577
9	Aram Restaurant	Asian Restaurant	Capitol building, DN Road	18.939543	72.834495
10	Kalpana restaurant	None	Gpo	18.937489	72.836157
11	Kamat's Restaurant	Indian Restaurant	Opp. Electric House, Colaba	18.921570	72.831004
12	Gulshan Restaurant	Indian Restaurant	NaN	18.938199	72.837764
13	Mangalore naaz Restaurant	Asian Restaurant	NaN	18.937245	72.837606
14	Sind Punjab Bar & Restaurant	Restaurant	263, SBS Road, Fort, Mumbai	18.937048	72.837381

Step5)We look for all the Hotels using the API. We fetch all the data and store it in a form of dataframe.

```
In [45]:
venues = results['response']['venues']

# tranform venues into a dataframe
dataframe1 = json_normalize(venues)
dataframe1.head()
```

```
C:\Users\deepa\anaconda3\lib\site-packages\ipykernel_launcher.py:4: FutureWarning: pandas.io.json.json_normalize is deprecated, use panda
s.json_normalize instead
  after removing the cwd from sys.path.
```

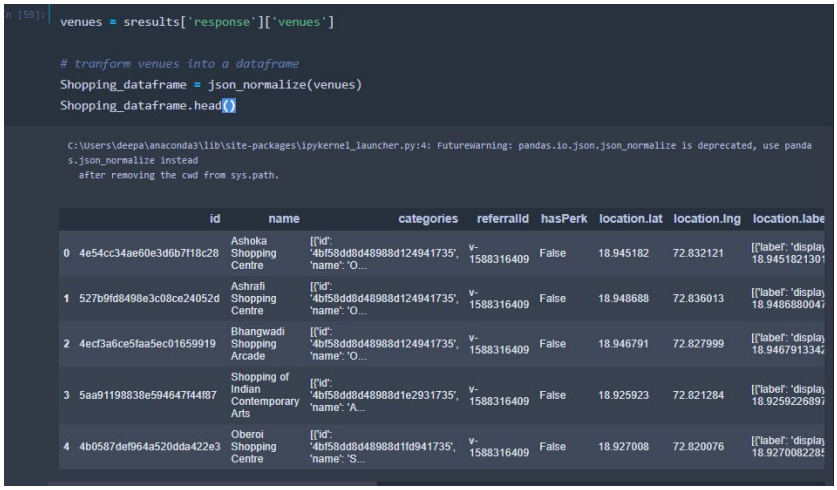
	id	name	categories	referralId	hasPerk	location.address	location.crossStreet	lo
0	4e135a03a809291902c7d32f	Hotel Traveller's Inn	[{'id': '4bf58dd8d48988d1f8931735', 'name': 'B...	v-1588315768	False	26, ADI Marzban Path	Behind Cafe Universal, Ballard Estate, Fort	18.
1	4bc6c48704e8b713f57f362d	Hotel Deluxe	[{'id': '4bf58dd8d48988d10f941735', 'name': 'I...	v-1588315768	False	Pitha Street	Fort	18.
2	4bcf4de40ffdce72215fb2c0	Residency Hotel	[{'id': '4bf58dd8d48988d1fa931735', 'name': 'H...	v-1588315768	False	26, Corner of D.N. Road and Rustom Sidhwa Marg...	NaN	18.
3	4cbc460f9552b60c8d81e48b	City Palace Hotel	[{'id': '4bf58dd8d48988d1fa931735', 'name': 'H...	v-1588315768	False	NaN	NaN	18.
4	4ffaf206e4b0a0306dd1271d	Hotel City Palace	[]	v-1588315768	False	Near CST	NaN	18.

Step6)We again check for any null values in our data and get rid of them.We also drop the columns which are of no use to us and hence clean our data.

```
In [50]:
clean_dataframe3 = clean_dataframe2.dropna(axis=0, how='any', thresh=None, subset=None, inplace=False)
clean_dataframe3
```

	name	categories	address	lat	lng
0	Hotel Traveller's Inn	Bed & Breakfast	26, ADI Marzban Path	18.936039	72.838000
1	Hotel Deluxe	Indian Restaurant	Pitha Street	18.933915	72.833205
2	Residency Hotel	Hotel	26, Corner of D.N. Road and Rustom Sidhwa Marg...	18.934978	72.833481
6	Hotel Fountain Plaza	Indian Restaurant	panthaky house, Raghunath dadaji street	18.935671	72.834030
7	Hotel Anand Bhuvan	Fast Food Restaurant	Opposite Shahid bhagat Singh Road	18.935345	72.836884
9	Hotel Landmark Fort	Bed & Breakfast	249 P D Mello Road	18.938528	72.837998
10	Colaba Grand Hotel	Hotel Bar	Colaba	18.939656	72.836457
15	Hotel Windsor	Hostel	Kumptha 10	18.936808	72.838327
18	Hotel Dolphin	Hotel	327,SBS Road	18.934567	72.836942
19	Welcome Hotel	Motel	257 Shahid Bhagat Singh Rd.	18.937703	72.837467
23	hotel golden palm fort	Hotel Pool	Fort	18.933103	72.834628
26	Hotel New Bengal	Hotel	Doctor D.N. Road	18.945722	72.833936
27	Hotel West End	Hotel	45 New Marine Lines	18.940741	72.827720
28	Hotel Madras Bhavan	Indian Restaurant	Kalbadevi	18.946648	72.828893
29	Deluxe Hotel	Housing Development	287 Shahid Bhagat Singh Road	18.935690	72.837398

Step7)We then look for all the shopping centres and fetch them using the API.We take the data and put it in a dataframe.



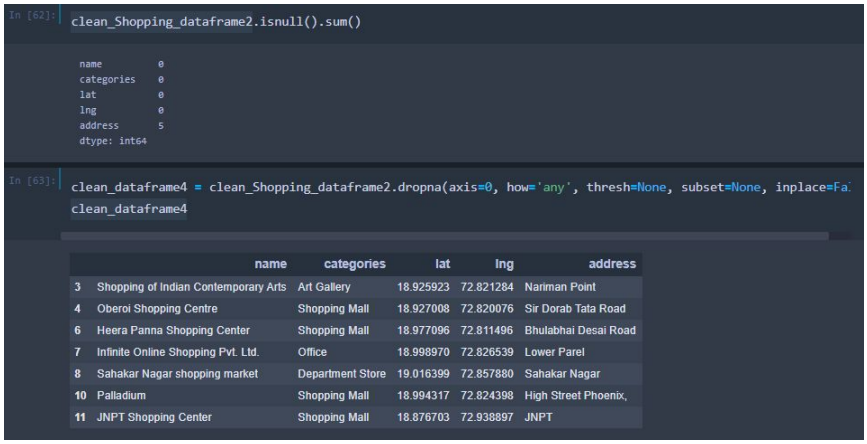
```
In [59]:
venues = sresults['response']['venues']

# tranform venues into a dataframe
Shopping_dataframe = json_normalize(venues)
Shopping_dataframe.head()
```

```
C:\Users\deepa\anaconda3\lib\site-packages\ipykernel_launcher.py:4: FutureWarning: pandas.io.json.json_normalize is deprecated, use panda
s.json_normalize instead
  after removing the cwd from sys.path.
```

	id	name	categories	referralId	hasPerk	location.lat	location.lng	location.labe
0	4e54cc34ae60e3d6b7f18c28	Ashoka Shopping Centre	[{'id': '4bf58dd8d48988d124941735', 'name': 'O...	v-1588316409	False	18.945182	72.832121	[{'label': 'displa 18.9451821301
1	527b9fd8498e3c08ce24052d	Ashrafi Shopping Centre	[{'id': '4bf58dd8d48988d124941735', 'name': 'O...	v-1588316409	False	18.948688	72.836013	[{'label': 'displa 18.948688004
2	4ecf3a6ce5faa5ec01659919	Bhangwadi Shopping Arcade	[{'id': '4bf58dd8d48988d124941735', 'name': 'O...	v-1588316409	False	18.946791	72.827999	[{'label': 'displa 18.946791334
3	5aa91198838e594647f44f87	Shopping of Indian Contemporary Arts	[{'id': '4bf58dd8d48988d1e2931735', 'name': 'A...	v-1588316409	False	18.925923	72.821284	[{'label': 'displa 18.925922689
4	4b0587def964a520dda422e3	Oberoi Shopping Centre	[{'id': '4bf58dd8d48988d1fd941735', 'name': 'S...	v-1588316409	False	18.927008	72.820076	[{'label': 'displa 18.927008228

Step8)We check for possible null values in our data and drop the columns that are not required in order to clean our data.



```
In [62]:
clean_Shopping_dataframe2.isnull().sum()
```

```
name          0
categories    0
lat           0
lng           0
address       5
dtype: int64
```

```
In [63]:
clean_dataframe4 = clean_Shopping_dataframe2.dropna(axis=0, how='any', thresh=None, subset=None, inplace=Fa
clean_dataframe4
```

	name	categories	lat	lng	address
3	Shopping of Indian Contemporary Arts	Art Gallery	18.925923	72.821284	Nariman Point
4	Oberoi Shopping Centre	Shopping Mall	18.927008	72.820076	Sir Dorab Tata Road
6	Heera Panna Shopping Center	Shopping Mall	18.977096	72.811496	Bhulabhai Desai Road
7	Infinite Online Shopping Pvt. Ltd.	Office	18.998970	72.826539	Lower Parel
8	Sahakar Nagar shopping market	Department Store	19.016399	72.857880	Sahakar Nagar
10	Palladium	Shopping Mall	18.994317	72.824398	High Street Phoenix,
11	JNPT Shopping Center	Shopping Mall	18.876703	72.938897	JNPT

Step9)We now have our three different cleaned data frames of all the restaurants,,shopping centres and hotels.We take these data frames and concatenate

them to form one single data frame.

```
In [65]: #dataframe of hotels, shopping stores and restaurants
         df = pd.concat([hotel_dataframe, df_Restaurant, clean_dataframe4], ignore_index=True)
         df
```

	name	categories	address	lat	lng
0	Hotel Traveller's Inn	Bed & Breakfast	26, ADI Marzban Path	18.936039	72.838000
1	Residency Hotel	Hotel	26, Corner of D.N. Road and Rustom Sidhwa Marg...	18.934978	72.833481
2	Hotel Landmark Fort	Bed & Breakfast	249 P D Mello Road	18.938528	72.837998
3	Colaba Grand Hotel	Hotel Bar	Colaba	18.939656	72.836457
4	Hotel Dolphin	Hotel	327,SBS Road	18.934567	72.836942
5	Welcome Hotel	Motel	257 Shahid Bhagat Singh Rd.	18.937703	72.837467
6	hotel golden palm fort	Hotel Pool	Fort	18.933103	72.834628
7	Hotel New Bengal	Hotel	Doctor D.N. Road	18.945722	72.833936
8	Hotel West End	Hotel	45 New Marine Lines	18.940741	72.827720
9	Farhang Restaurant	Steakhouse	Opposite GPO	18.938165	72.837917
10	Nanumal Bhojraj Restaurant - Fort	Indian Restaurant	Near GPO, Fort, VT	18.937974	72.837663
11	Iran Like Restaurant	Indian Restaurant	Palton Road, Near Haj House	18.945343	72.836809
12	New Majestic Restaurant	Indian Restaurant	Opp CST	18.938972	72.835517
13	Stadium Restaurant & Stores	Indian Restaurant	Veer Nariman Road	18.933173	72.826929
14	Panchratna restaurant & bar	Indian Restaurant	Opp. Metro Cinema	18.942900	72.826820
15	Nanumal Bhojraj Restaurant - Masjid Bunder	Indian Restaurant	Gaumukh Bhawan, Masjid Bunder West, Near Masji...	18.953200	72.837577
16	Aram Restaurant	Asian Restaurant	Capitol building, DN Road	18.939543	72.834495
17	Kamat's Restaurant	Indian Restaurant	Opp. Electric House, Colaba	18.921570	72.831004

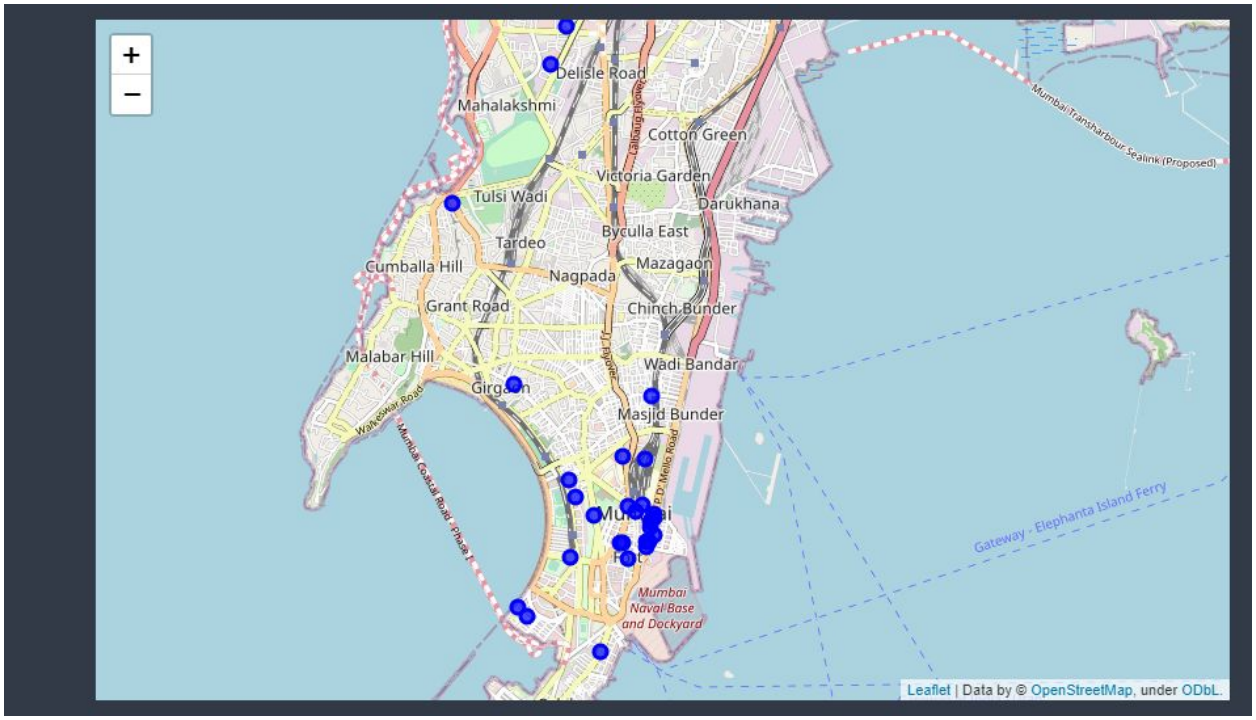
Step10) We make use of folium library to make the map of the region and use various methods to mark our locations on the map to distinguish them from other places.

```
In [68]: df_map = folium.Map(location=[latitude, longitude], zoom_start=14)

         for lat, lng, name, categories, address in zip(df['lat'], df['lng'], df['name'],df['categories'],df['addres
             label = '{}, {}'.format(name, address)
             label = folium.Popup(label, parse_html=True)
             folium.CircleMarker(
                 [lat, lng],
                 radius=5,
                 popup=label,
                 color='blue',
                 fill=True,
                 fill_color='blue',
                 fill_opacity=0.7,
                 parse_html=False).add_to(df_map)

         df_map #mapping all of these on a map
```

Step11) The final map after doing the visualization looks like this.



RESULT-

We were able to successfully create a map visualization to show all the nearby shopping centres,Restaurants and Hotels which might help the tourists who are visiting the city for the first time, to get good food, tourism and overall a great experience..This Project was build to help the tourists find the best and most comfortable stay possible in the city.

CONCLUSION-

There is always room for improvement and hence the above solution I have provided can also be improved for best results depending upon the data we have.